Assignment #9

Jingye Wang

November 8, 2016

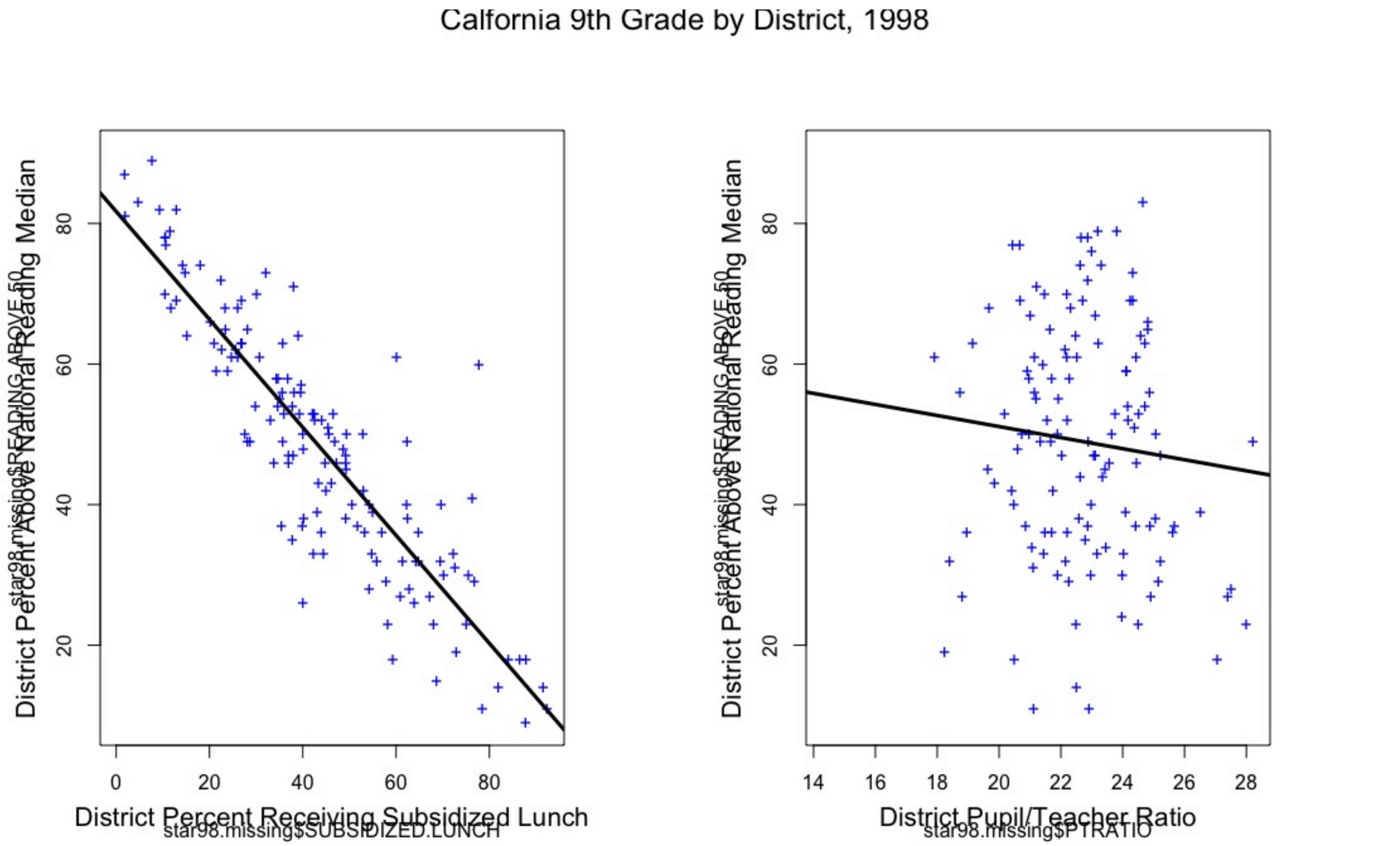
Excercise 1

This problem deals with missing data in a linear model.

Part 1

First download and graph the data using:

```
star98.missing <- read.table("star98.missing.dat_.txt",header=TRUE)
par(mfrow=c(1,2),mar=c(5,5,5,5))
plot(star98.missing$SUBSIDIZED.LUNCH,star98.missing$READING.ABOVE.50,pch="+",col="blue")
abline(lm(star98.missing$READING.ABOVE.50~star98.missing$SUBSIDIZED.LUNCH),lwd=3)
mtext(side=1,cex=1.3,line=2.5,"District Percent Receiving Subsidized Lunch")
mtext(side=2,cex=1.3,line=2.5,"District Percent Above National Reading Median")
plot(star98.missing$PTRATIO,star98.missing$READING.ABOVE.50,pch="+",col="blue")
abline(lm(star98.missing$READING.ABOVE.50~star98.missing$PTRATIO),lwd=3)
mtext(side=1,cex=1.3,line=2.5,"District Pupil/Teacher Ratio")
mtext(side=2,cex=1.3,line=2.5,"District Percent Above National Reading Median")
mtext(side=3,cex=1.5,outer=TRUE,line=-1,"Calfornia 9th Grade by District, 1998")
```



Part 2

Determine how much missing data there is and if there is a discernable pattern.

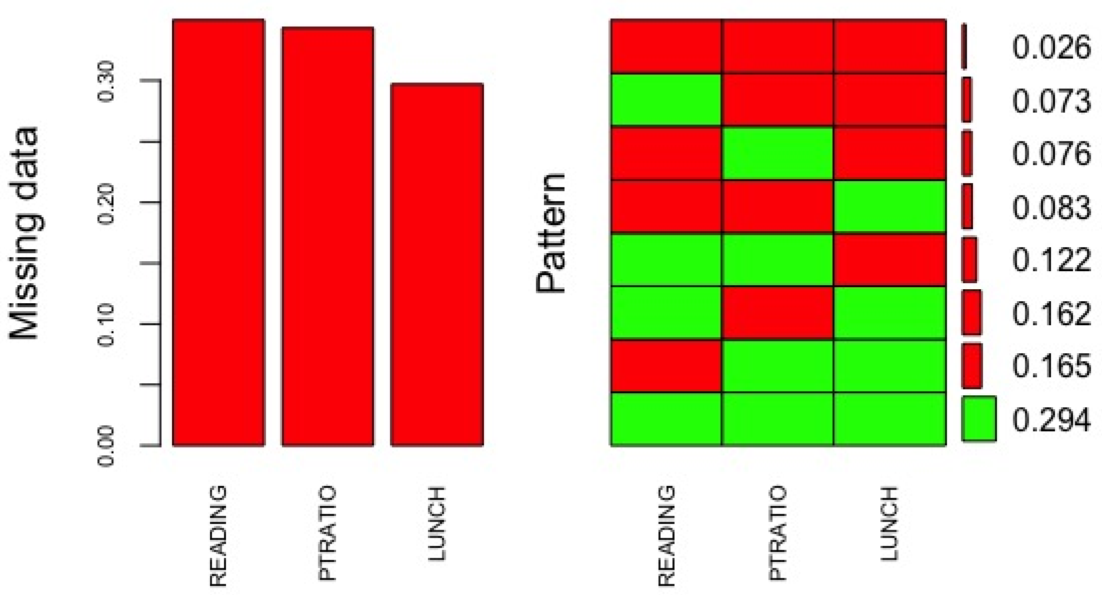
```
colnames(star98.missing) <- c('LUNCH','PTRATIO','READING')
test_missing <- apply(star98.missing, 2, function(x) sum(is.na(x)))
test_missing
```

```
SUBSIDIZED.LUNCH         PTRATIO    READING.ABOVE.50
              90             104                 106
```

Based on the result, SUBSIDIZED.LUNCH has 90 missing values, PTRATIO has 104 missing values, and READING.ABOVE.50 has 106 missing values.

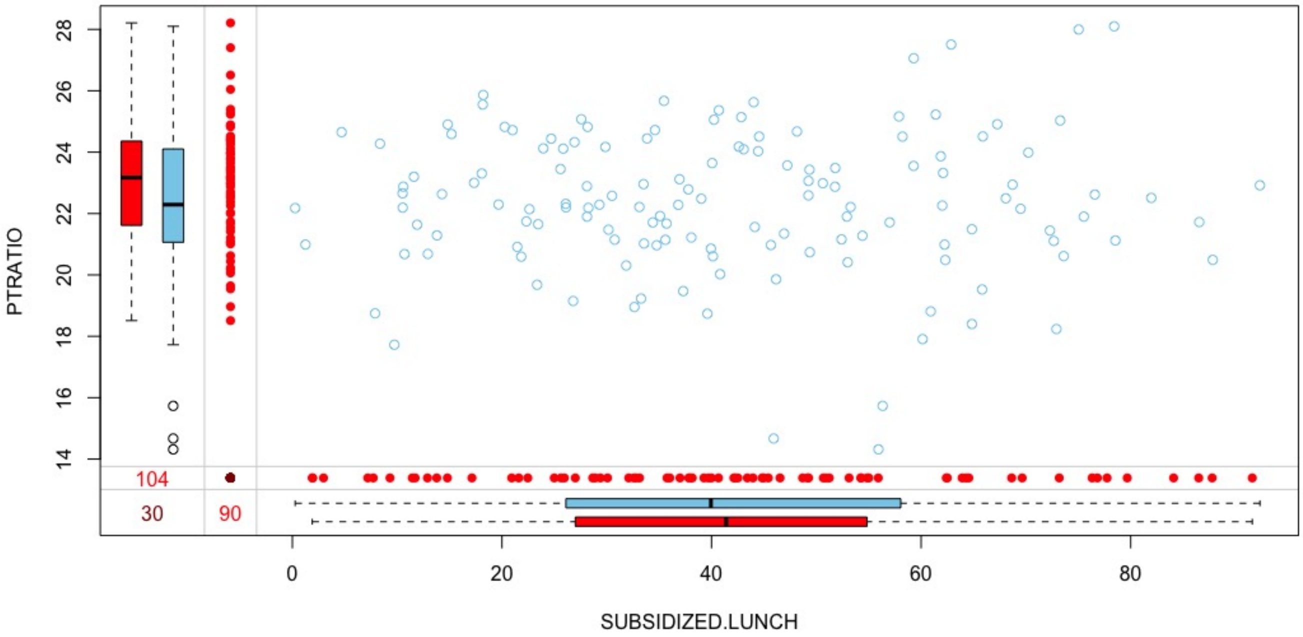
```
library(VIM)
missing_plot <- aggr(star98.missing, col=c('green','red'),
                  numbers=TRUE, sortVars=TRUE,
                  labels=names(star98.missing), cex.axis=.7,
                  gap=3, ylab=c("Missing data","Pattern"))
```

```
Variables sorted by number of missings:
 Variable     Count
  READING 0.3498350
  PTRATIO 0.3432343
    LUNCH 0.2970297
```

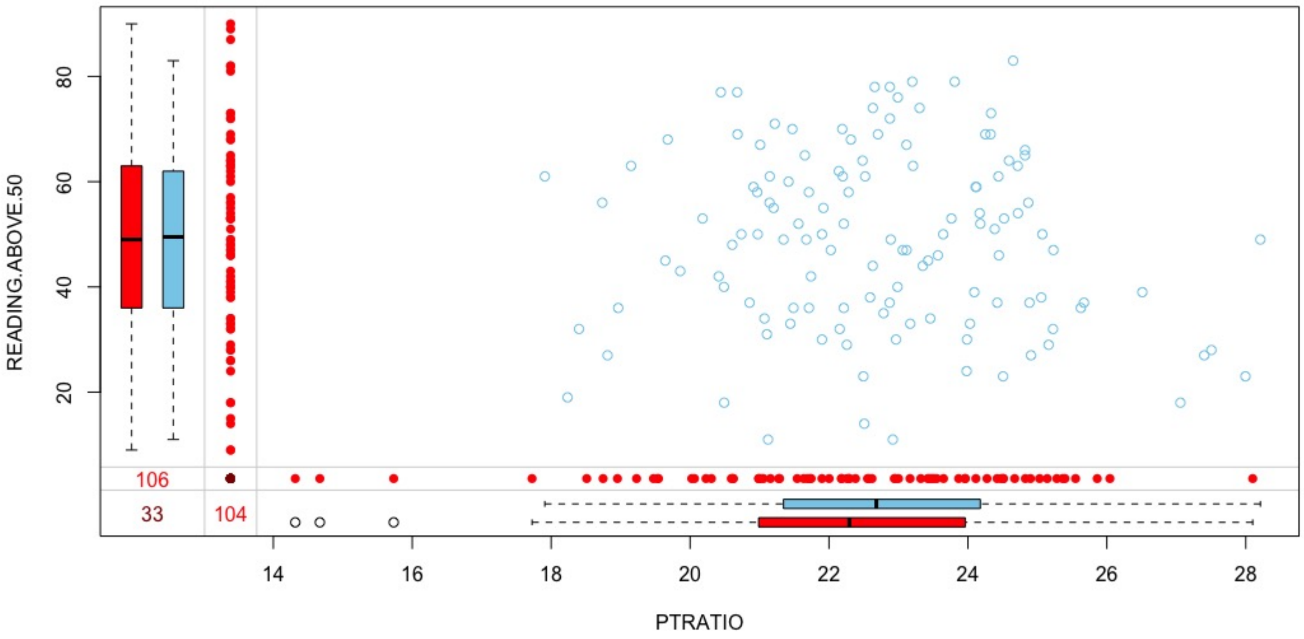


The plot helps us understanding that almost 29.4% of the samples are not missing any information, 34.9% are missing the READING.ABOVE.50 value, 34.3% are missing the PTRATIO value, and 29.7% are missing the SUBSIDIZED.LUNCH value.

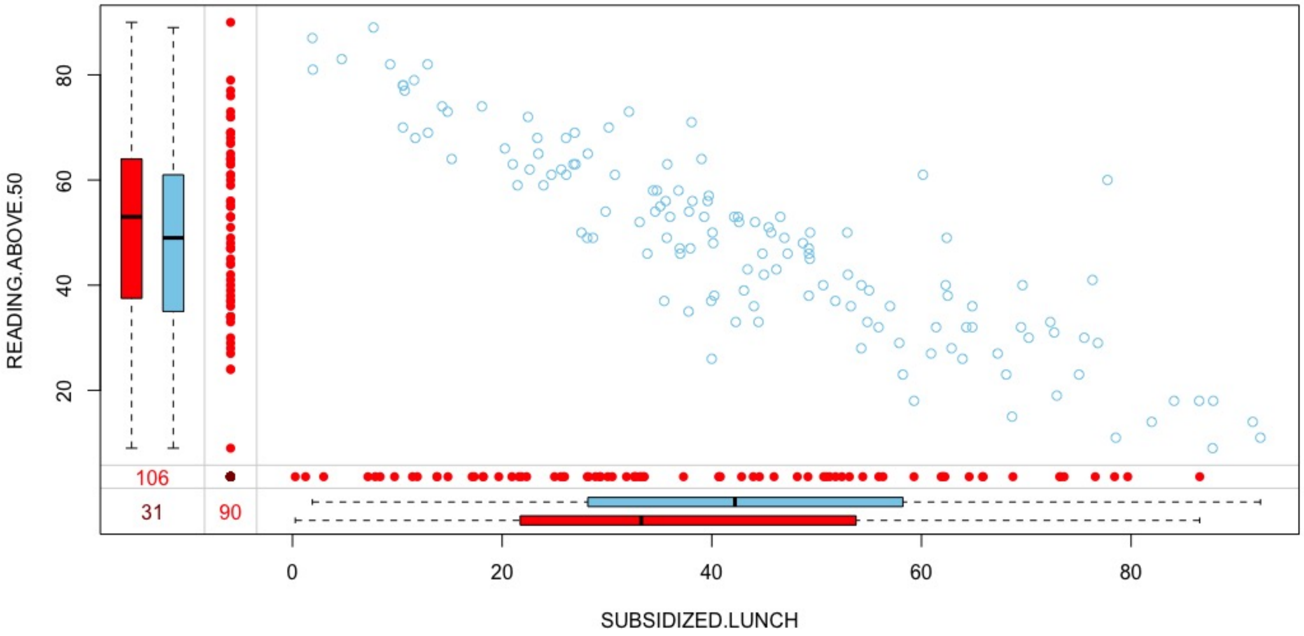
```
marginplot(star98.missing[,1:2])
marginplot(star98.missing[,2:3])
marginplot(star98.missing[,c(1,3)])
```



The red box plot on the left shows the distribution of PTRATIO with SUBSIDIZED.LUNCH missing while the blue box plot shows the distribution of the remaining datapoints. We can see that the missing data of SUBSIDIZED.LUNCH has a higher mean of PTRATIO than the non-missing data. The missing data of PTRATIO has a higher mean of SUBSIDIZED.LUNCH than the non-missing data.



The missing data of READING.ABOVE.50 has the same mean of PTRATIO than the non-missing data. The missing data of PTRATIO has a lower mean of READING.ABOVE.50 than the non-missing data.



The missing data of READING.ABOVE.50 has a higher mean of SUBSIDIZED.LUNCH than the non-missing data. The missing data of SUBSIDIZED.LUNCH has a lower mean of READING.ABOVE.50 than the non-missing data. READING.ABOVE.50 and SUBSIDIZED.LUNCH shows a linear corelation with each other.

Part 3

Now use mice to run a new model. Also run a model omitting cases with missing data. What differences do you observe? Which is better?

```
star98.missing.imp <- mice(star98.missing,m=10)
star98.complete <- complete(star98.missing.imp,1)
star98.omit <- na.omit(star98.missing)
```

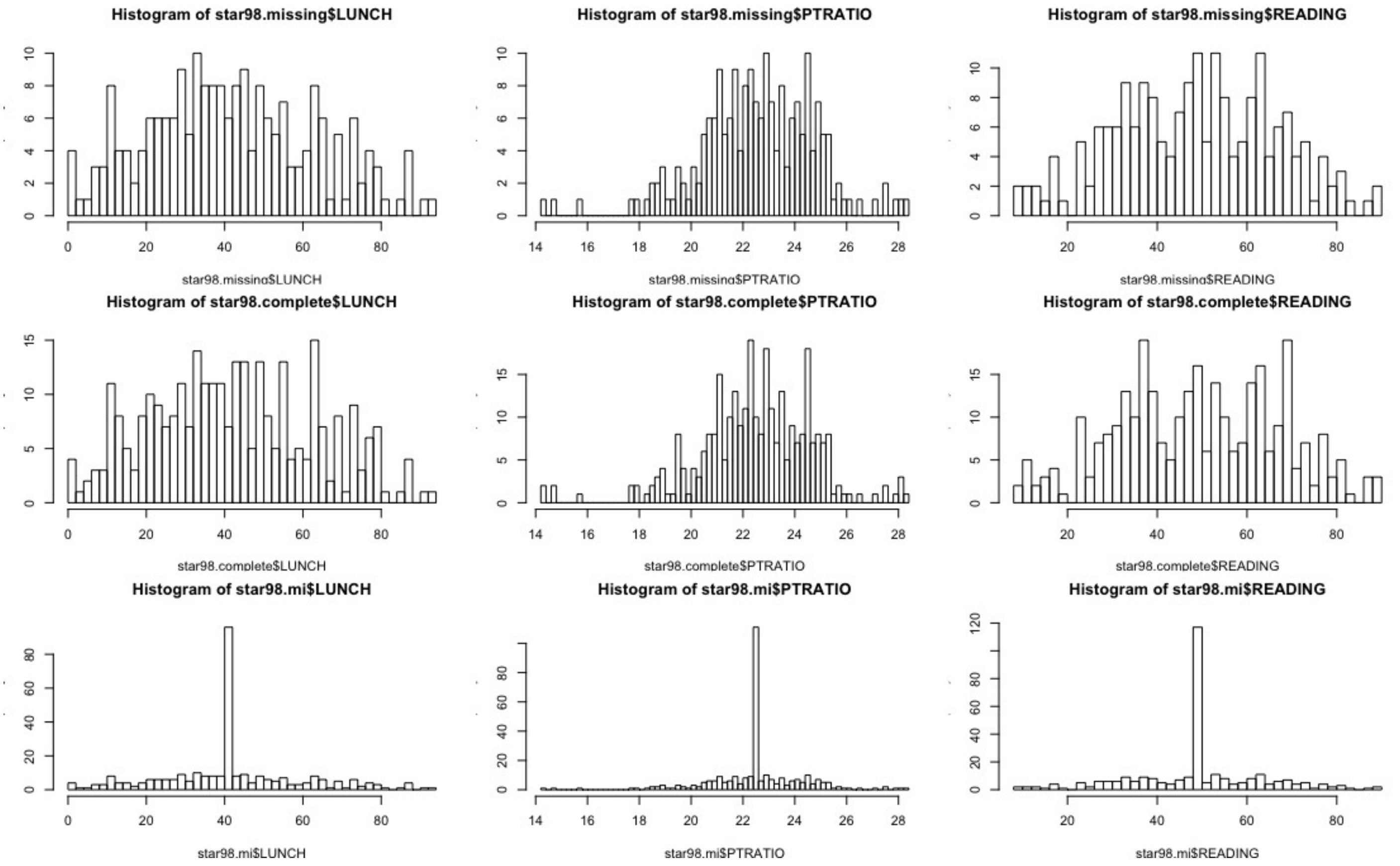
The complete data made by mice is better than omit missing data. Because after na.omit, the number of observations change from 303 to 89, which will lose many information.

```
star98.mi <- mi(star98.missing)

# Distribution distort
par(mfrow=c(3,3))
hist(star98.missing$LUNCH, breaks = 50)
hist(star98.missing$PTRATIO, breaks = 50)
hist(star98.missing$READING, breaks = 50)

hist(star98.complete$LUNCH, breaks = 50)
hist(star98.complete$PTRATIO, breaks = 50)
hist(star98.complete$READING, breaks = 50)

hist(star98.mi$LUNCH, breaks = 50)
hist(star98.mi$PTRATIO, breaks = 50)
hist(star98.mi$READING, breaks = 50)
```



Histogram of star98.missing$LUNCH

Histogram of star98.missing$PTRATIO

Histogram of star98.missing$READING

Histogram of star98.complete$LUNCH

Histogram of star98.complete$PTRATIO

Histogram of star98.complete$READING

Histogram of star98.mi$LUNCH

Histogram of star98.mi$PTRATIO

Histogram of star98.mi$READING

Based on this figure, replace NAs with mean is not a good way for the imputation. Because all NAs within one column will be the same number, which is impossible in the complete data and the shape of the data is very strange.

```
# Standard deviation

sapply(star98.missing, function(x) sd(x, na.rm = T))
    LUNCH   PTRATIO   READING 
21.283941  2.247492 17.802918 
sapply(star98.complete, function(x) sd(x, na.rm = T))
    LUNCH   PTRATIO   READING 
20.950600  2.220446 18.142250 
sapply(star98.mi, function(x) sd(x, na.rm = T))
    LUNCH   PTRATIO   READING 
17.832676  1.819815 14.342200 
```

The standard deviations in the data which is replaced NAs with mean become much smaller, which will make some fake results when we try to do clustering or other analysis.

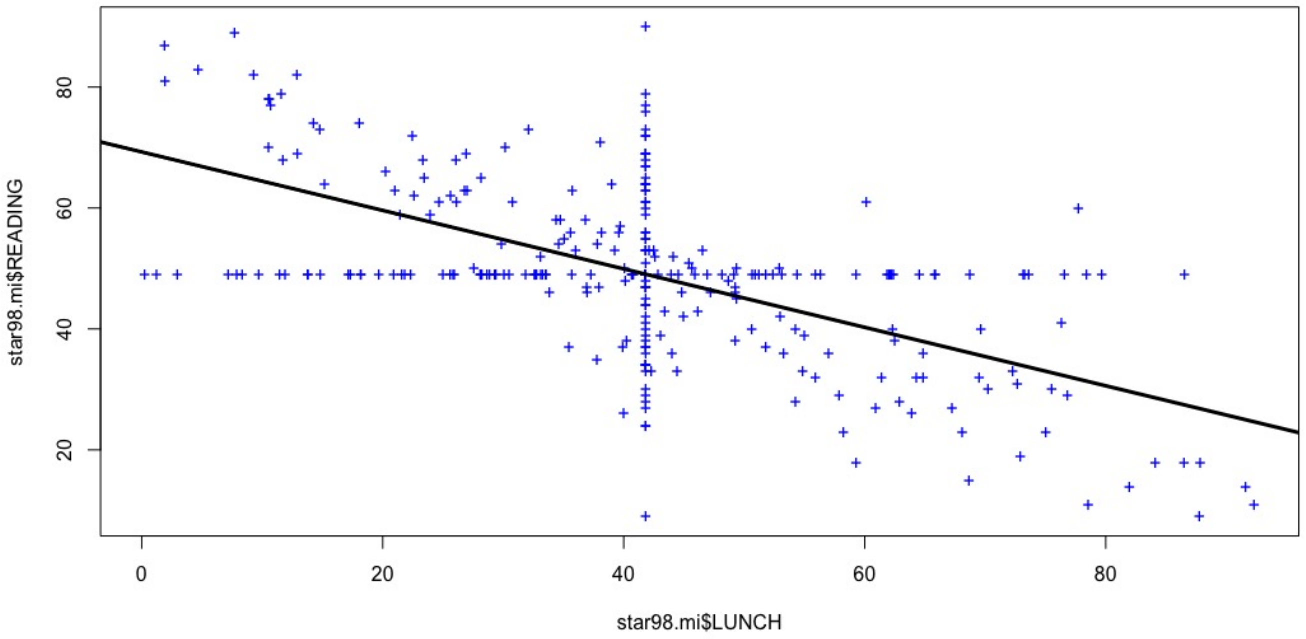
```
# Correlation

cor(na.omit(star98.missing))
              LUNCH     PTRATIO     READING
LUNCH    1.00000000 -0.08192835 -0.90208782
PTRATIO -0.08192835  1.00000000 -0.09996259
READING -0.90208782 -0.09996259  1.00000000

cor(star98.complete)
              LUNCH     PTRATIO    READING
LUNCH    1.00000000  0.02024835 -0.8791195
PTRATIO  0.02024835  1.00000000 -0.2019051
READING -0.87911948 -0.20190513  1.0000000

cor(star98.mi)
               LUNCH      PTRATIO     READING
LUNCH    1.000000000 -0.005889029 -0.60158948
PTRATIO -0.005889029  1.000000000 -0.05148008
READING -0.601589481 -0.051480081  1.00000000

par(mfrow=c(1,1))
plot(star98.mi$LUNCH,star98.mi$READING,pch="+",col="blue")
abline(lm(star98.mi$READING ~ star98.mi$LUNCH),lwd=3)
```



If we replace NAs with mean, it will decrease the corelations between variables.

```
model <- lm(READING.ABOVE.50~SUBSIDIZED.LUNCH,data=star98.complete)
summary(model)
```

```
Call:
lm(formula = READING.ABOVE.50 ~ SUBSIDIZED.LUNCH, data = star98.complete)

Residuals:
    Min      1Q  Median      3Q     Max 
-25.893  -5.193   0.086   4.053  41.811 

Coefficients:
                 Estimate Std. Error t value Pr(>|t|)    
(Intercept)      80.82870    1.15036   70.26   <2e-16 ***
SUBSIDIZED.LUNCH -0.72342    0.02458  -29.43   <2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 8.899 on 301 degrees of freedom
Multiple R-squared:  0.7421,    Adjusted R-squared:  0.7413 
F-statistic: 866.2 on 1 and 301 DF,  p-value: < 2.2e-16
```

```
model_omit <- lm(READING.ABOVE.50~SUBSIDIZED.LUNCH,data=star98.omit)
summary(model_omit)
```

```
Call:
lm(formula = READING.ABOVE.50 ~ SUBSIDIZED.LUNCH, data = star98.omit)

Residuals:
     Min       1Q   Median       3Q      Max 
-16.9576  -4.1594   0.4502   4.7060  26.7118 

Coefficients:
                 Estimate Std. Error t value Pr(>|t|)    
(Intercept)      81.70887    1.90336   42.93   <2e-16 ***
SUBSIDIZED.LUNCH -0.78843    0.04044  -19.50   <2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 7.569 on 87 degrees of freedom
Multiple R-squared:  0.8138,    Adjusted R-squared:  0.8116 
F-statistic: 380.1 on 1 and 87 DF,  p-value: < 2.2e-16
```

Based on these two models, the coefficiency and the intercept are different from each other. Thus, missing value imputation need to be thought carefully before data analysis.

Excercise 2

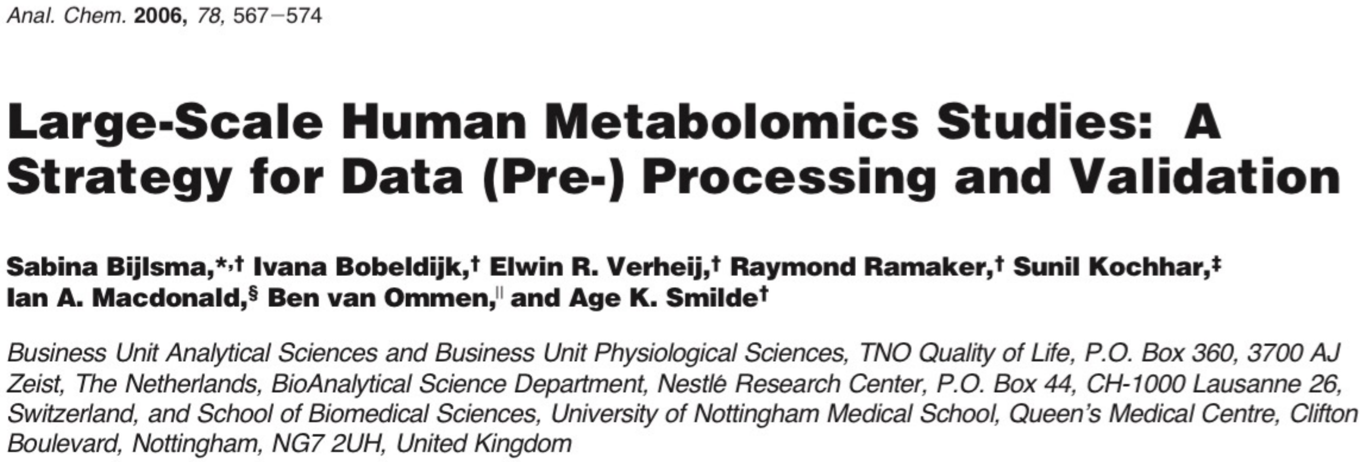
Explain what the following R does and why you would not want to do this.

```
mi <- function(data.mat) {
  for (i in 1:ncol(data.mat)) {
    if (sum(is.na(data.mat[,i])) > 0) {
      print(paste("column",i,"has missing data"))
      mean.col <- mean(data.mat[,i],na.rm=TRUE)
    for (j in 1:nrow(data.mat)) {
      if (is.na(data.mat[j,i]) ==TRUE) data.mat[j,i] <- mean.col }
    }
  }
  return(data.mat)
}
```

This function first check whether it has missing values for each column. And then this function replaces missing values of each column with the mean of the column.

Excercise 3

Find an article in your literature that uses case-wise deletion. Discuss how you might replicate the model and improve the work.



Anal. Chem. **2006,** *78,* 567−574

Large-Scale Human Metabolomics Studies: A Strategy for Data (Pre-) Processing and Validation

Sabina Bijlsma,[*,†] Ivana Bobeldijk,[†] Elwin R. Verheij,[†] Raymond Ramaker,[†] Sunil Kochhar,[‡] Ian A. Macdonald,[§] Ben van Ommen,[||] and Age K. Smilde[†]

Business Unit Analytical Sciences and Business Unit Physiological Sciences, TNO Quality of Life, P.O. Box 360, 3700 AJ Zeist, The Netherlands, BioAnalytical Science Department, Nestlé Research Center, P.O. Box 44, CH-1000 Lausanne 26, Switzerland, and School of Biomedical Sciences, University of Nottingham Medical School, Queen's Medical Centre, Clifton Boulevard, Nottingham, NG7 2UH, United Kingdom

This paper has been cited for 456 times, which is also treated as a "golden rule" when solve missing values in metabolomics data. In this paper, they said "If a variable had a nonzero measurement value in at least 80% of the variables, the variable was included in the data set; otherwise the variable was removed." I don't agree with this. In the field of metabolomics, the reason for missing values are the concentration of compounds is lower than the detection limits of the machine, which is very commonly observed in metabolomics data analyses. If we use case-wise deletion, there will be a huge bias of the results. I will use the censored-data model using Bugs to do the missing value imputation on page 405 in Gelman&Hill book.